

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2004-056605

(43)Date of publication of application : 19.02.2004

(51)Int.Cl.

H04N 1/41

H03M 7/30

H03M 7/40

H04N 7/30

(21)Application number : 2002-213368

(71)Applicant : NIPPON HOSO KYOKAI <NHK>

(22)Date of filing : 23.07.2002

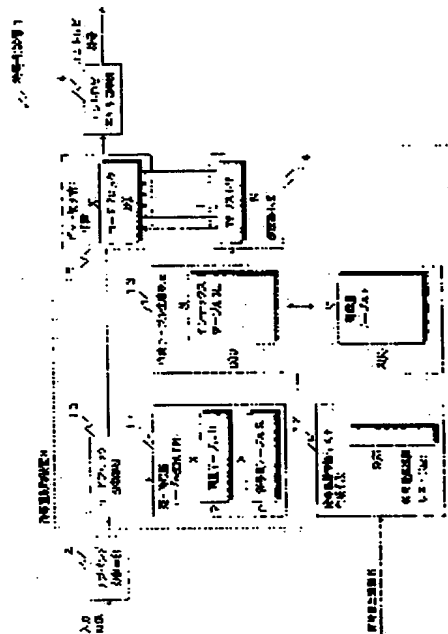
(72)Inventor : KUGE TETSUO

(54) METHOD, DEVICE AND PROGRAM FOR CONTROLLING CODE AMOUNT

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a code amount controlling method, a code amount controller and a code amount control program for calculating a bit position at which distortion quantity becomes minimum with a small amount of operation when an image is divided into blocks of a particular size and bit positions for discarding the bit plane of each of the blocks are determined in encoding the image under the upper limit value of a code amount.

SOLUTION: This code amount controller 3 is provided with a code block dividing means 10, a distortion/code amount table generating means 11, a code amount reference value list generating means 12, a work table generating means 13, a reverse searching means 14 and a bit omitting means 15. In addition, a bit position at which the total distortion quantity between a first block and a particular block is minimum and the total distortion amount at that time is generated as a work table for every value of a plurality of code amount reference values where a code amount upper limit value is an upper limit. and a bit position where the total distortion amount of the whole blocks is minimum is reversely searched from the work table.



LEGAL STATUS

[Date of request for examination]

07.02.2005

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision
of rejection]

[Date of requesting appeal against examiner's
decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2004-56605

(P2004-56605A)

(43) 公開日 平成16年2月19日(2004.2.19)

(51) Int.Cl. ⁷	F I	テーマコード (参考)
HO4N 1/41	HO4N 1/41 B	5C059
HO3M 7/30	HO3M 7/30 A	5C078
HO3M 7/40	HO3M 7/40	5J064
HO4N 7/30	HO4N 7/133 Z	

審査請求 未請求 請求項の数 5 O L (全 24 頁)

(21) 出願番号 特願2002-213368 (P2002-213368)
 (22) 出願日 平成14年7月23日 (2002.7.23)

(71) 出願人 000004352
 日本放送協会
 東京都渋谷区神南2丁目2番1号
 (74) 代理人 100064414
 弁理士 磯野 道造
 (72) 発明者 久下 哲郎
 東京都世田谷区砧一丁目10番11号
 日本放送協会 放送技術研究

所内

F ターム (参考) 5C059 KK22 KK30 MA00 MA24 MC11
 MC38 ME01 PP01 TA36 TB08
 TC18 TC38 TD06 TD15 UA02
 5C078 AA04 BA53 CA21 DA01
 5J064 AA01 AA02 BA09 BA16 BC01
 BC16 BC26 BD01

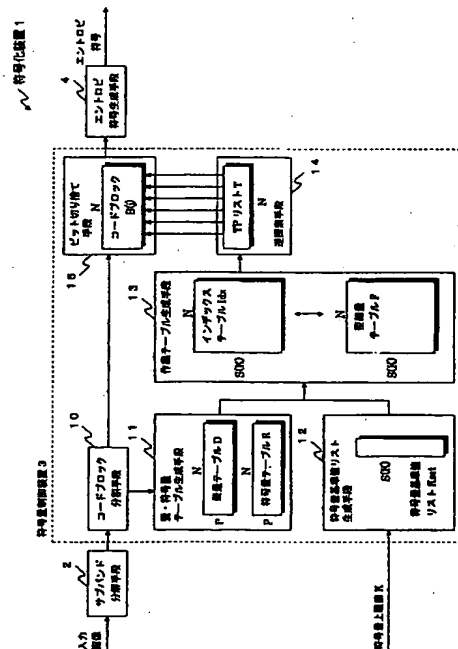
(54) 【発明の名称】 符号量制御方法、符号量制御装置及び符号量制御プログラム

(57) 【要約】

【課題】 画像を符号量上限値の下で符号化する際に、その画像を特定の大きさに分割し、その分割された各ブロックのビットプレーンを廃棄するビット位置を決定するときに、歪量が最小となるビット位置を少ない演算量で求める符号量制御方法、符号量制御装置及び符号量制御プログラムを提供する。

【解決手段】 符号量制御装置3は、コードブロック分割手段10と、歪・符号量テーブル生成手段11と、符号量基準値リスト生成手段12と、作業テーブル生成手段13と、逆探索手段14と、ビット切り捨て手段15と、を備え、符号量上限値を上限とする複数の符号量基準値毎に、第1番目のブロックから特定ブロックまでの歪総量を最小とするビット位置及びそのときの歪総量を作業テーブルとして生成し、その作業テーブルから全ブロックの歪総量を最小とするビット位置を逆探索することとを特徴とする。

【選択図】 図1



【特許請求の範囲】

【請求項 1】

符号化対象となる画像を特定の大きさのブロックに分割して、その分割されたブロックの順に可変長符号を生成する画像符号化において、与えられた符号量上限値の範囲内で前記画像を符号化するために、前記各ブロックの符号量を制御する符号量制御方法であって、前記ブロックを量子化する複数の量子化レベルに基づいて前記ブロックを符号化したときの符号量と、前記量子化レベルに基づいて前記ブロックから廃棄される歪量とを算出し、歪・符号量テーブルとして生成する歪・符号量テーブル生成ステップと、前記符号量上限値を最大値とする複数の符号量基準値からなる符号量基準値リストを生成する符号量基準値リスト生成ステップと、
 第 1 番目のブロックで前記符号量基準値に最も近い符号量となる前記量子化レベル及びそのときの歪量を、前記符号量基準値と前記ブロックとを対応付けたインデックステーブル及び歪総量テーブルの初期値として設定する作業テーブル初期値設定ステップと、前記符号量基準値毎に、前記歪総量テーブルを参照して、第 1 番目のブロックから特定ブロックまでの歪総量を最小とする量子化レベル及びそのときの歪総量を順次決定して、第 2 番目以降のブロックにおける前記インデックステーブル及び前記歪総量テーブルの値を設定する作業テーブル設定ステップと、
 この作業テーブル設定ステップで設定した前記インデックステーブルから、前記全ブロックの歪総量を最小とする量子化レベルを逆探索して求める逆探索ステップと、
 を含んでいることを特徴とする符号量制御方法。

10

20

【請求項 2】

符号化対象となる画像を特定の大きさのブロック $B(j)$ ($j = 1, \dots, N$) に分割して、そのブロック毎に可変長符号を生成する画像符号化において、与えられた符号量上限値 K の範囲内で前記画像を符号化するために、前記各ブロックの符号量を制御する符号量制御方法であって、
 前記ブロックを量子化する複数の量子化レベル t に基づいて前記ブロック $B(j)$ を符号化したときの符号量 $R(t, j)$ と、前記量子化レベル t に基づいて前記ブロック $B(j)$ から廃棄される歪量 $D(t, j)$ とを算出し、歪・符号量テーブルとして生成する歪・符号量テーブル生成ステップと、
 前記符号量上限値 K を最大値とする $S(K)$ 個の符号量基準値 $k(i)$ ($i = 1, \dots, S(K) = K$) からなる符号量基準値リストを生成する符号量基準値リスト生成ステップと

30

、
 前記符号量基準値 $k(i)$ に最も近い符号量 $R(t, 1)$ となる前記量子化レベル t 及びそのときの歪量 $D(t, 1)$ を、前記符号量基準値 $k(i)$ と前記ブロック $B(j)$ とを対応付けたインデックステーブル $Idx(i, j)$ 及び歪総量テーブル $F(i, j)$ の初期値として設定する作業テーブル初期値設定ステップと、
 符号量を r とし、その r に最も近い前記符号量基準値 $k(i)$ のインデックス i を返す写像を $\psi(r)$ とし、歪総量 $f(i, j; t)$ を $D(t, j) + F(\psi(k(i)) - R(t, j))$ 、 $j - 1$ と定義したときの、前記歪総量 $f(i, j; t)$ を最小とする前記量子化レベル t 及びそのときの前記歪総量 $f(i, j; t)$ を、インデックステーブル $Idx(i, j)$ 及び歪総量テーブル $F(i, j)$ に設定する作業テーブル設定ステップと

40

、
 前記符号量基準値 $k(i)$ のインデックス i に $S(K)$ を、前記ブロック $B(j)$ のインデックス j に N を初期値として与え、前記ブロック $B(j)$ に割り当てる量子化レベル $t(j)$ をインデックステーブル $Idx(i, j)$ とし、前記写像 $\psi(k(i) - R(t(j), j))$ を計算し、その計算結果を改めて i の値とし、 j を順次減算することで、前記量子化レベル $t(j)$ のリスト $(t(1), \dots, t(N))$ を前記インデックステーブル $Idx(i, j)$ から逆探索する逆探索ステップと、
 を含んでいることを特徴とする符号量制御方法。

【請求項 3】

50

前記量子化レベル t は、前記各ブロック $B(j)$ のビットプレーンの位置を示す値であり、そのビットプレーンの位置に基づいて各ブロック $B(j)$ を量子化することを特徴とする請求項2に記載の符号量制御方法。

【請求項4】

符号化対象となる画像を特定の大きさのブロックに分割して、その分割されたブロックの順に可変長符号を生成する画像符号化において、与えられた符号量上限値の範囲内で前記画像を符号化するために、前記各ブロックの符号量を制御する符号量制御装置であって、前記ブロックを量子化する複数の量子化レベルに基づいて前記ブロックを符号化したときの符号量と、前記量子化レベルに基づいて前記ブロックから廃棄される歪量とを算出し、歪・符号量テーブルとして生成する歪・符号量テーブル生成手段と、

この歪・符号量テーブル生成手段によって生成された前記歪・符号量テーブルを保持する歪・符号量テーブル保持手段と、

前記符号量上限値を最大値とする複数の符号量基準値からなる符号量基準値リストを生成する符号量基準値リスト生成手段と、

この符号量基準値リスト生成手段で生成された符号量基準値リストを保持する符号量基準値リスト保持手段と、

第1番目のブロックから特定ブロックまでの符号量が、前記符号量基準値に最も近い符号量となる前記特定ブロックの量子化レベル及びそのときの歪総量を、前記符号量基準値と前記ブロックとを対応付けたインデックステーブル及び歪総量テーブルとして生成する作業テーブル生成手段と、

この作業テーブル生成手段で生成された前記インデックステーブル及び前記歪総量テーブルを保持する作業テーブル保持手段と、

この作業テーブル保持手段に保持されている前記インデックステーブルから、前記全ブロックの歪総量を最小とする量子化レベルを逆探索して求める逆探索手段と、

を備えていることを特徴とする符号量制御装置。

【請求項5】

符号化対象となる画像を特定の大きさのブロックに分割して、その分割されたブロックの順に可変長符号を生成する画像符号化において、与えられた符号量上限値の範囲内で前記画像を符号化するときの前記各ブロックの符号量を制御するために、コンピュータを、前記ブロックを量子化する複数の量子化レベルに基づいて前記ブロックを符号化したときの符号量と、前記量子化レベルに基づいて前記ブロックから廃棄される歪量とを算出し、歪・符号量テーブルとして生成して歪・符号量テーブル保持手段に保持する歪・符号量テーブル生成手段、

前記符号量上限値を最大値とする複数の符号量基準値からなる符号量基準値リストを生成して符号量基準値リスト保持手段に保持する符号量基準値リスト生成手段、

第1番目のブロックから特定ブロックまでの符号量が、前記符号量基準値に最も近い符号量となる前記量子化レベル及びそのときの歪総量を、前記符号量基準値と前記ブロックとを対応付けたインデックステーブル及び歪総量テーブルを生成して作業テーブル保持手段に保持する作業テーブル生成手段、

前記作業テーブル保持手段で保持されている前記インデックステーブルから、前記全ブロックの歪総量を最小とする量子化レベルを逆探索して求める逆探索手段、

として機能させることを特徴とする符号量制御プログラム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、画像符号化技術に関し、より詳細には、符号量を制限された条件の下で、符号化データの切り捨てを行い画像を圧縮符号化する際に、歪量を最も少なくして符号化を行うための符号量制御を行う符号量制御方法、符号量制御装置及び符号量制御プログラムに関する。

【0002】

10

20

30

40

50

【従来の技術】

従来、符号量を制限された条件の下で、符号化データの切り捨てを行い画像を圧縮符号化する符号化方式としては、ウェーブレット (Wavelet) 変換により符号化を行う J P E G 2 0 0 0 (J o i n t P h o t o g r a p h i c E x p e r t G r o u p 2 0 0 0) 等が挙げられる。

【0003】

この J P E G 2 0 0 0 は、入力画像を、ウェーブレット変換を用いた 2 次元サブバンド分解によって分解し、その分解された各サブバンドの画素を複数ビット (P ビット) で表現し、さらに、各サブバンドを空間的に特定の大きさ (垂直 m 画素、水平 n 画素) のブロック (コードブロック) に分割する。そして、各ブロック毎に $m \times n$ ビットの P 個のビットプレーンが生成されることになる。J P E G 2 0 0 0 は、このビットプレーンを 1 単位として独立してエントロピ符号化を行うことで、圧縮符号化を行うものである。例えば、入力画像がブロック数 N に分解されていたとすると、 $N \times P$ 個のビットプレーンが生成されることになり、入力画像は、J P E G 2 0 0 0 によって $N \times P$ 個のエントロピ符号に変換 (符号化) されることになる。

【0004】

ここで、図 11 を参照して、J P E G 2 0 0 0 等で入力画像を 2 次元サブバンド分解によって分解し、各サブバンドをコードブロックに分割する例について説明する。図 11 (a) は、入力画像を 2 次元ウェーブレット変換により、2 次元空間周波数サブバンドに分解した例を示している。図 11 (b) は、サブバンドを特定の大きさのコードブロックに分割した例を示している。図 11 (a) に示すように、2 次元ウェーブレット変換は、入力画像に 1 次元のウェーブレット変換を水平及び垂直に施すものであり、1 回の分解によって入力画像を $L1L1$ 、 $L1H1$ 、 $H1L1$ 、 $H1H1$ に分解する。この分解をさらに $L1L1$ に対して行うことで、 $L1L1$ は、 $L2L2$ 、 $L2H2$ 、 $H2L2$ 、 $H2H2$ に分解される。

【0005】

このように低周波領域を繰り返して分解することで、2 次元ウェーブレット変換は、入力画像の階層的なサブバンド分解を実現している。なお、サブバンド分解を行う目的は、符号化された単一の符号化コードストリームに対して、受信側でその一部分を切り出すだけで低解像度の画像を再現することを可能にするためである。そして、図 11 (a) のように分解されたサブバンドを、図 11 (b) に示すように特定の大きさのコードブロックに分割する。

【0006】

このようにコードブロックに分割された入力画像は、ビットプレーン単位で符号化を行うことで、画像全体の符号化が行われる。この画像全体の符号量は、画像毎に異なるため、一定符号量以下に符号量を制限するには、画像全体として歪量が最小限になるようにビットプレーンを選択的に廃棄することが行われている。このビットプレーンを選択的に廃棄して符号化を行うことをビットプレーン符号化という。

【0007】

ここで、図 12 を参照して、J P E G 2 0 0 0 で用いられているビットプレーン (bit plane) 符号化の動作について説明する。図 12 は、入力画像をコードブロックに分割し、そのコードブロック毎のビットプレーンを選択的に廃棄して全体の符号量を一定符号量以下に抑えるビットプレーン符号化を模式的に表した図である。

図 12 に示すように、入力画像 Img は、 N 個のコードブロック B ($B(j)$ ($j = 1, \dots, N$)) に分割される。なお、個々のコードブロック B は、 $m \times n$ ビットの P 個のビットプレーンで構成される。

【0008】

ここでコードブロック B の中で着目するビット位置を t ($t = 0, \dots, P - 1$) とし、 t より下位のビットプレーンを廃棄してコードブロック B を符号化したときの、符号量 R を (1) 式で定義する。

10

20

30

40

50

【0009】

$$R(t, j) \equiv : r(t, j) + \dots + r(P-1, j) \quad \dots (1)$$

【0010】

なお、 $r(t, j)$ は、コードブロック $B(j)$ ($j = 1, \dots, N$) において、ビットプレーン t をエントロピ符号化したときの符号量を示す。すなわち、符号量 R は、コードブロック B のビットプレーン t からビットプレーン $(P-1)$ までの符号量を示す。この着目するビット位置 t は、ビットプレーンを廃棄する基準点を指し示すので、以下 TP (Truncation Point) と呼ぶ。

また、(1) 式において廃棄された歪量 D を (2) 式で定義する。

【0011】

$$D(t, j) \equiv : b(0, j) * 2^0 + \dots + b(t-1, j) * 2^{t-1} \quad \dots (2)$$

【0012】

なお、 $b(t, j)$ は、コードブロック $B(j)$ ($j = 1, \dots, N$) において、ビットプレーン t の非ゼロビットの総数を示す。すなわち、歪量 D は、コードブロック B のビットプレーン 0 からビットプレーン $(t-1)$ までの歪量の和を示す。この歪量 D は、例えば、 $b(t, j) * 2^t$ 以外にも、その自乗値を用いて算出することとしてもよい。

【0013】

一般に、サブバンド分解によって分解されたサブバンド (係数) は正負の値を取り得るので、最上位のビットプレーン $(P-1)$ を係数の符号を示すビットプレーン (係数符号プレーン) とし、ビットプレーン $0, \dots, P-2$ を係数の絶対値のビット値とする。

【0014】

例えば、 TP が $t = 0$ の場合、ビットプレーンを一切廃棄しないので、(2) 式において、 $D(0, j) = 0$ となる。また、 $TP = P-1$ の場合、ビットプレーンを全て廃棄するので、(1) 式において、 $R(P-1, j) = 0$ となる。

【0015】

ここで、コードブロック B のビットプレーンの数を 16 ($P = 16$) としたときの TP と、符号量 $R(t)$ 及び歪量 $D(t)$ との関係を図 13 に示す。図 13 では、ビットプレーンの TP の位置 $t = 15$ を係数符号プレーンとしている。ここでビットプレーンの TP を t としたとき、 t より下位のビットプレーン $0, \dots, t-1$ の t 個のビットプレーンが廃棄される。また、 $r(s)$ ($0 \leq s \leq 15$) は、ビットプレーン s をエントロピ符号化したときの符号量を示し、 $d(s)$ ($0 \leq s \leq 15$) は、ビットプレーン s を廃棄するときの歪量を表している。この歪量は、例えば (ビットプレーン s の非ゼロビットの総数) $* 2^s$ 、あるいはその自乗値とする。また、 $R(t)$ は、ビットプレーンの TP の位置を t としたときのコードブロック $B(j)$ ($j = 1, \dots, N$) 全体の符号量を示し、 $D(t)$ は、コードブロック $B(j)$ 全体の歪量を示している。

【0016】

さらに、図 14 にビットプレーンの TP の位置 (大きさ) と歪量 $D(t)$ 及び符号量 $R(t)$ との関係を、あるコードブロック B についてモデル的に示す。図 14 (a) はビットプレーンの TP の位置 (大きさ) と歪量 $D(t)$ との関係を、図 14 (b) はビットプレーンの TP の位置 (大きさ) と符号量 $R(t)$ との関係を示している。図 14 に示すように、 TP が大きくなる (上位のビットプレーンから多くのビットプレーンが廃棄される) と、歪量 $D(t)$ が増大し、符号量 $R(t)$ が減少する。逆に、 TP が小さくなる (下位のビットプレーンで少ないビットプレーンが廃棄される) と、歪量 $D(t)$ が減少し、符号量 $R(t)$ が増大する。

ここで、コードブロック $B(j)$ ($j = 1, \dots, N$) について、コードブロックの TP の位置を示す $t(j)$ のリスト (TP リスト) T を (3) 式で定義する。

【0017】

$$T \equiv (t(1), \dots, t(N)) \quad \dots (3)$$

【0018】

すなわち、コードブロック $B(j)$ の TP の位置をビットプレーン $t(j)$ とすると、 N

10

20

30

40

50

個のコードブロック全体の符号量総和 $R(T)$ は (4) 式で、歪量総和 $D(T)$ は、(5) 式で表すことができる。

【0019】

$$R(T) = R(t(1), 1) + \dots + R(t(N), N) \quad \dots (4)$$

【0020】

$$D(T) = D(t(1), 1) + \dots + D(t(N), N) \quad \dots (5)$$

【0021】

これにより、J P E G 2 0 0 0 等で符号量を制限された条件の下で、符号化データの切り捨てを行い画像を圧縮符号化するには、以下の(問題A)の解を求めればよいことになる。

10

【0022】

(問題A)

目標とする符号量上限値を K とするとき、 $R(T) = K$ なる条件の下で、 $D(T)$ を最小化する TP (Truncation Point) のリスト (TP リスト) T を決定する (T 、 $R(T)$ 及び $D(T)$ は (3) 式、(4) 式及び (5) 式参照)。

(問題A終)

【0023】

この(問題A)の解を直接求めることは膨大な数の組合せ演算を要するため、従来、J P E G 2 0 0 0 では、(問題A)の類似的な問題として以下の(問題B)の解を求め、リスト (TP リスト) T ((3) 式) を決定している。

20

【0024】

(問題B)

目標とする符号量上限値を K とするとき、 $R(T) = K$ なる条件の下で、(6) 式で示した評価関数を最小化する組合せ T と λ を求める。

【0025】

$$L(T, \lambda) \equiv R(T) + \lambda D(T) \quad \dots (6)$$

【0026】

(T 、 $R(T)$ 及び $D(T)$ は (3) 式、(4) 式及び (5) 式参照)

(問題B終)

【0027】

この(問題B)の解を求めるには、一般的には評価関数 $L(T, \lambda)$ が連続関数の場合であれば、ラグランジュ (L a g r a n g e) の未定乗数法を用いて解析的に解くことが可能である。しかし、評価関数 $L(T, \lambda)$ は離散関数であるため、予め設定した複数の λ の値により、その λ の値を変化させることで、評価関数 $L(T, \lambda)$ をより小さくする TP のリスト $T(t(1), \dots, t(N))$ を逐次的に決定する等の近似的な手法が用いられている。

30

【0028】

このように、従来では、近似的な手法により、 TP のリスト T を決定し、各コードブロック毎に $t(1), \dots, t(N)$ より下位のビットプレーンを廃棄し、各ビットプレーンを符号化することで、符号量を制限された条件の下で、画像の圧縮符号化を行っている。

40

【0029】

【発明が解決しようとする課題】

しかし、前記従来の技術では、(問題A)の解は(問題B)の解となるが、(問題B)の解は本来求めたい(問題A)の解となる保証がない。すなわち、(問題B)の解は、近似的な手法により TP (Truncation Point) のリスト $T(t(1), \dots, t(N))$ を求めるため、評価関数 $L(T, \lambda)$ ((6) 式参照) を最小にする最適解にならない場合があるという問題がある。

【0030】

なお、(問題A)の解を直接求める方法としては、全検索により解を求める方法がある。すなわち、 N 個のコードブロック $B(j)$ ($j = 1, \dots, N$) において、ある TP のリス

50

ト T に対する $R(T)$ 及び $D(T)$ ((4) 式及び (5) 式参照) を求め、逐次リスト T を変化させることで、 $R(T) = K$ (K は符号量上限値) を満たし、 $D(T)$ が最小となるリスト T を探索する方法である。

【0031】

しかし、この方法では、例えば、水平 512 画素、垂直 512 画素で 1 画素 8 ビットの画像について、コードブロックを 64 に分割して符号化を行う場合、すなわち、画像を水平 64 画素、垂直 64 画素の 64 個のコードブロック ($N=64$) に分割し、各コードブロックのビットプレーンの数が 8 ($P=8$) であるとする、取り得るリスト T の数は $P^N = 6.2 \times 10^{57}$ となり、1 つのリスト T の検証を 1 ナノ秒 (ns) で行ったとしても、 $D(T)$ が最小となるリスト T を探索するには、 6.2×10^{48} 秒 (sec) $= 2 \times 10^{41}$ 年を要することになり、現実的な時間では検索することができない。

10

【0032】

本発明は、以上のような問題点に鑑みてなされたものであり、符号量を制限された条件の下で、符号化データの切り捨てを行い画像を圧縮符号化する場合に、その画像を特定の大きさのブロック (コードブロック) に分割し、各ブロックのビットプレーンを廃棄するビット位置を決定する際に、歪量が最小となる最適なビット位置を全検索よりも遥かに少ない演算量で求めることを可能にする符号量制御方法、符号量制御装置及び符号量制御プログラムを提供することを目的とする。

【0033】

【課題を解決するための手段】

本発明は、前記目的を達成するために創案されたものであり、まず、請求項 1 に記載の符号量制御方法は、符号化対象となる画像を特定の大きさのブロックに分割して、その分割されたブロックの順に可変長符号を生成する画像符号化において、与えられた符号量上限値の範囲内で画像を符号化するために、各ブロックの符号量を制御する符号量制御方法であって、以下のステップを含むものとした。

20

【0034】

すなわち、ブロックを量子化する複数の量子化レベルに基づいてブロックを符号化したときの符号量と、そのときの量子化レベルに基づいて前記ブロックから廃棄される歪量とを算出し、歪・符号量テーブルとして生成する歪・符号量テーブル生成ステップと、符号量上限値を最大値とする複数の符号量基準値からなる符号量基準値リストを生成する符号量基準値リスト生成ステップと、第 1 番目のブロックで符号量基準値に最も近い符号量となる量子化レベル及びそのときの歪量を、符号量基準値とブロックとを対応付けたインデックステーブル及び歪総量テーブルの初期値として設定する作業テーブル初期値設定ステップと、符号量基準値毎に、歪総量テーブルを参照して、第 1 番目のブロックから特定ブロックまでの歪総量を最小とする量子化レベル及びそのときの歪総量を順次決定して、第 2 番目以降のブロックにおけるインデックステーブル及び歪総量テーブルの値を設定する作業テーブル設定ステップと、この作業テーブル設定ステップで設定したインデックステーブルから、全ブロックの歪総量を最小とする量子化レベルを逆探索して求める逆探索ステップと、を含むことを特徴とする。

30

【0035】

この方法によれば、符号量制御方法は、歪・符号量テーブル生成ステップによって、複数の量子化レベルに基づいてブロックを符号化したときの符号量と、そのときの量子化レベルに基づいてブロックから廃棄される歪量とを算出し、歪・符号量テーブルとして生成する。この量子化レベルは、符号化を行う際の符号量を削減するためのレベルを示すものである。

40

【0036】

そして、符号量制御方法は、符号量基準値リスト生成ステップによって、符号量上限値を最大値とする複数の符号量基準値からなる符号量基準値リストを生成する。この符号量基準値は、符号量上限値における最適な量子化レベルを求める問題を解くときに、小規模な問題を解決し、その結果から順次問題の規模を拡大することで求める問題を解くために、

50

符号量上限値を小規模な問題の値になるように細分化したものである。例えば、H D T V (High Definition Television) の入力画像を符号化する場合は、符号量基準値の個数を100~200個程度とする。

【0037】

さらに、符号量制御方法は、作業テーブル初期値設定ステップによって、第1番目のブロックで符号量基準値に最も近い符号量となる前記量子化レベル及びそのときの歪量を、符号量基準値とブロックとを対応付けたインデックステーブル及び歪総量テーブルの初期値として設定し、作業テーブル設定ステップによって、符号量基準値毎に、歪総量テーブルを参照して、第1番目のブロックから特定ブロックまでの歪総量を最小とする量子化レベル及びそのときの歪総量を順次決定して、第2番目以降のブロックにおけるインデックス
10
ステータブル及び歪総量テーブルの値を設定する。これによって、作業テーブル初期値設定ステップで初期化したインデックステーブル及び歪総量テーブルは、作業テーブル初期値設定ステップによって、帰納的にブロックの順に値が設定される。

【0038】

そして、符号量制御方法は、逆探索ステップによって、作業テーブル設定ステップで設定したインデックステーブルから、全ブロックの歪総量を最小とする量子化レベルを逆探索して求める。この逆探索ステップでは、すでに作業テーブル設定ステップでインデックステーブルが設定されているため、このインデックステーブルを参照することで、歪総量を最小とする量子化レベルを最終ブロックから順番に検索していき、全ブロックの量子化レ
20
ベルを取得する。

【0039】

また、請求項2に記載の符号量制御方法は、符号化対象となる画像を特定の大きさのブロックB(j) (j=1, ..., N) に分割して、そのブロック毎に可変長符号を生成する画像符号化において、与えられた符号量上限値Kの範囲内で画像を符号化するために、前記各ブロックの符号量を制御する符号量制御方法であって、以下のステップを含むものとした。

【0040】

すなわち、ブロックを量子化する複数の量子化レベルtに基づいてブロックB(j)を符号化したときの符号量R(t, j)と、量子化レベルtに基づいてブロックB(j)から廃棄される歪量D(t, j)とを算出し、歪・符号量テーブルとして生成する歪・符号量
30
テーブル生成ステップと、符号量上限値Kを最大値とするS(K)個の符号量基準値k(i) (i=1, ..., S(K)=K) からなる符号量基準値リストを生成する符号量基準値リスト生成ステップと、符号量基準値k(i) に最も近い符号量R(t, 1)となる量子化レベルt及びそのときの歪量D(t, 1)を、符号量基準値k(i)とブロックB(j)とを対応付けたインデックステーブルIdx(i, j)及び歪総量テーブルF(i, j)の初期値として設定する作業テーブル初期値設定ステップと、を含む。

【0041】

さらに、符号量をrとし、そのrに最も近い前記符号量基準値k(i)のインデックスiを返す写像を $\psi(r)$ として、歪総量 $f(i, j; t)$ を $D(t, j) + F(\psi(k(i) - R(t, j)), j - 1)$ と定義したときの、歪総量 $f(i, j; t)$ を最小とする
40
量子化レベルt及びそのときの歪総量 $f(i, j; t)$ を、インデックステーブルIdx(i, j)及び歪総量テーブルF(i, j)に設定する作業テーブル設定ステップと、符号量基準値k(i)のインデックスiにS(K)を、ブロックB(j)のインデックスjにNを初期値として与え、ブロックB(j)に割り当てる量子化レベルt(j)をインデックステーブルIdx(i, j)とし、写像 $\psi(k(i) - R(t(j), j))$ を計算し、その計算結果を改めてiの値とし、jを順次減算することで、量子化レベルt(j)のリスト(t(1), ..., t(N))を前記インデックステーブルIdx(i, j)から逆探索する逆探索ステップと、を含むことを特徴とする。

【0042】

この方法によれば、符号量制御方法は、歪・符号量テーブル生成ステップによって、プロ
50

ックを量子化する複数の量子化レベル t に基づいてブロック $B(j)$ を符号化したときの符号量 $R(t, j)$ と、量子化レベル t に基づいてブロック $B(j)$ から廃棄される歪量 $D(t, j)$ とを算出し、歪・符号量テーブルとして生成する。

【0043】

そして、符号量制御方法は、符号量基準値リスト生成ステップによって、符号量基準値リストを生成する。この符号量基準値リストは、符号量上限値 K を最大値とする $S(K)$ 個の符号量基準値 $k(i)$ ($i = 1, \dots, S(K) = K$) からなる。

【0044】

さらに、符号量制御方法は、作業テーブル初期値設定ステップによって、符号量基準値 $k(i)$ とブロック $B(j)$ とを対応付けたインデックステーブル $Idx(i, j)$ 及び歪総量テーブル $F(i, j)$ を初期化する。この初期化で使用する初期値は、符号量基準値 $k(i)$ に最も近い符号量 $R(t, 1)$ となる量子化レベル t 及びそのときの歪量 $D(t, 1)$ とする。そして、作業テーブル設定ステップによって、インデックステーブル $Idx(i, j)$ 及び歪総量テーブル $F(i, j)$ に値を設定する。このインデックステーブル $Idx(i, j)$ 及び歪総量テーブル $F(i, j)$ に設定する値は、写像 $\psi(r)$ を符号量 r に最も近い符号量基準値 $k(i)$ のインデックス i を返す写像とし、歪総量 $f(i, j; t)$ を $D(t, j) + F(\psi(k(i) - R(t, j)), j - 1)$ と定義したときの、歪総量 $f(i, j; t)$ を最小とする量子化レベル t 及びそのときの歪総量 $f(i, j; t)$ とする。なお、写像 $\psi(r)$ は、符号量 r と符号量基準値 $k(i)$ との差分を求め、その差分量が最も小さくなるインデックス i を返すものである。

【0045】

そして、符号量制御方法は、逆探索ステップによって、量子化レベル $t(j)$ のリスト ($t(1), \dots, t(N)$) をインデックステーブル $Idx(i, j)$ から逆探索する。この逆探索は、符号量基準値 $k(i)$ のインデックス i に $S(K)$ を、ブロック $B(j)$ のインデックス j に N を初期値として与え、ブロック $B(j)$ に割り当てる量子化レベル $t(j)$ をインデックステーブル $Idx(i, j)$ とし、写像 $\psi(k(i) - R(t(j), j))$ を計算し、その計算結果を改めて i の値とし、 j を順次減算することで行う。

【0046】

さらに、請求項3に記載の符号量制御方法は、請求項2に記載の符号量制御方法であって、量子化レベル t は、各ブロック $B(j)$ のビットプレーンの位置を示す値であり、そのビットプレーンの位置に基づいて各ブロック $B(j)$ を量子化することを特徴とする。

【0047】

この方法によれば、符号量制御方法は、量子化レベル t を各ブロック $B(j)$ のビットプレーンの位置を示す値とし、この値に基づいて、ビットプレーンの廃棄を行うことで、各ブロック $B(j)$ の量子化を行う。これによって、符号化における符号量を削減する。

【0048】

また、請求項4に記載の符号量制御装置は、符号化対象となる画像を特定の大きさのブロックに分割して、その分割されたブロックの順に可変長符号を生成する画像符号化において、与えられた符号量上限値の範囲内で画像を符号化するために、各ブロックの符号量を制御する符号量制御装置であって、以下の構成にかかるものとした。

【0049】

すなわち、ブロックを量子化する複数の量子化レベルに基づいてブロックを符号化したときの符号量と、量子化レベルに基づいてブロックから廃棄される歪量とを算出し、歪・符号量テーブルとして生成する歪・符号量テーブル生成手段と、この歪・符号量テーブル生成手段によって生成された歪・符号量テーブルを保持する歪・符号量テーブル保持手段と、符号量上限値を最大値とする複数の符号量基準値からなる符号量基準値リストを生成する符号量基準値リスト生成手段と、この符号量基準値リスト生成手段で生成された符号量基準値リストを保持する符号量基準値リスト保持手段と、第1番目のブロックから特定ブロックまでの符号量が、符号量基準値に最も近い符号量となる特定ブロックの量子化レベル及びそのときの歪総量を、符号量基準値とブロックとを対応付けたインデックステー

10

20

30

40

50

ル及び歪総量テーブルとして生成する作業テーブル生成手段と、この作業テーブル生成手段で生成されたインデックステーブル及び歪総量テーブルを保持する作業テーブル保持手段と、この作業テーブル保持手段に保持されているインデックステーブルから、全ブロックの歪総量を最小とする量子化レベルを逆探索して求める逆探索手段と、を備える構成とした。

【0050】

かかる構成によれば、符号量制御装置は、歪・符号量テーブル生成手段によって、複数の量子化レベルに基づいてブロックを符号化したときの符号量と、量子化レベルに基づいてブロックから廃棄される歪量とを算出し、歪・符号量テーブルとして生成し歪・符号量テーブル保持手段に保持する。この量子化レベルは、符号化を行う際の符号量を削減するためのレベルを示すものである。

10

【0051】

そして、符号量制御装置は、符号量基準値リスト生成手段によって、符号量上限値を最大値とする複数の符号量基準値からなる符号量基準値リストを生成し符号量基準値リスト保持手段に保持する。この符号量基準値は、符号量上限値における最適な量子化レベルを求める問題を解くときに、小規模な問題を解決し、その結果から順次問題の規模を拡大することで求める問題を解くために、符号量上限値を小規模な問題の値とするために細分化したものである。例えば、HDTV (High Definition Television) の入力画像を符号化する場合は、符号量基準値の個数を100～200個程度とする。

20

【0052】

さらに、符号量制御装置は、作業テーブル生成手段によって、第1番目のブロックから特定ブロックまでの符号量が、符号量基準値に最も近い符号量となる特定ブロックの量子化レベル及びそのときの歪総量を、符号量基準値とブロックとを対応付けたインデックステーブル及び歪総量テーブルとして生成し作業テーブル保持手段に保持する。

【0053】

そして、符号量制御装置は、逆探索手段によって、作業テーブル保持手段に保持されているインデックステーブルから、全ブロックの歪総量を最小とする量子化レベルを逆探索して求める。この逆探索手段では、すでに作業テーブル生成手段によって、インデックステーブルが作業テーブル保持手段に保持されているため、このインデックステーブルを参照することで、歪総量を最小とする量子化レベルを最終ブロックから順番に検索していき、全ブロックの量子化レベルを取得する。

30

【0054】

さらに、請求項5に記載の符号量制御プログラムは、符号化対象となる画像を特定の大きさのブロックに分割して、その分割されたブロックの順に可変長符号を生成する画像符号化において、与えられた符号量上限値の範囲内で画像を符号化するときの各ブロックの符号量を制御するために、コンピュータを、以下の手段によって機能させる構成とした。

【0055】

すなわち、ブロックを量子化する複数の量子化レベルに基づいてブロックを符号化したときの符号量と、量子化レベルに基づいてブロックから廃棄される歪量とを算出し、歪・符号量テーブルとして生成して歪・符号量テーブル保持手段に保持する歪・符号量テーブル生成手段、符号量上限値を最大値とする複数の符号量基準値からなる符号量基準値リストを生成して符号量基準値リスト保持手段に保持する符号量基準値リスト生成手段、第1番目のブロックから特定ブロックまでの符号量が、符号量基準値に最も近い符号量となる前記量子化レベル及びそのときの歪総量を、符号量基準値とブロックとを対応付けたインデックステーブル及び歪総量テーブルを生成して作業テーブル保持手段に保持する作業テーブル生成手段、作業テーブル保持手段で保持されているインデックステーブルから、全ブロックの歪総量を最小とする量子化レベルを逆探索して求める逆探索手段、とした。

40

【0056】

かかる構成によれば、符号量制御プログラムは、歪・符号量テーブル生成手段によって、

50

複数の量子化レベルに基づいてブロックを符号化したときの符号量と、量子化レベルに基づいてブロックから廃棄される歪量とを算出し、歪・符号量テーブルとして生成し歪・符号量テーブル保持手段に保持する。この量子化レベルは、符号化を行う際の符号量を削減するためのレベルを示すものである。

【0057】

そして、符号量制御プログラムは、符号量基準値リスト生成手段によって、符号量上限値を最大値とする複数の符号量基準値からなる符号量基準値リストを生成し符号量基準値リスト保持手段に保持する。この符号量基準値は、符号量上限値における最適な量子化レベルを求める問題を解くときに、小規模な問題を解決し、その結果から順次問題の規模を拡大することで求める問題を解くために、符号量上限値を小規模な問題の値とするために細分化したものである。例えば、HDTV (High Definition Television) の入力画像を符号化する場合は、符号量基準値の個数を100～200個程度とする。

10

【0058】

さらに、符号量制御プログラムは、作業テーブル生成手段によって、第1番目のブロックから特定ブロックまでの符号量が、符号量基準値に最も近い符号量となる量子化レベル及びそのときの歪総量を、符号量基準値とブロックとを対応付けたインデックステーブル及び歪総量テーブルを生成し作業テーブル保持手段に保持する。

【0059】

そして、符号量制御プログラムは、逆探索手段によって、作業テーブル保持手段に保持されているインデックステーブルから、全ブロックの歪総量を最小とする量子化レベルを逆探索して求める。この逆探索手段では、すでに作業テーブル生成手段によって、インデックステーブルが作業テーブル保持手段に保持されているため、このインデックステーブルを参照することで、歪総量を最小とする量子化レベルを最終ブロックから順番に検索していき、全ブロックの量子化レベルを取得する。

20

【0060】

【発明の実施の形態】

以下、本発明の実施の形態について図面を参照して説明する。

図1は、本発明における符号量制御装置を含んだ符号化装置の構成を示したブロック図である。図1に示す符号化装置1は、サブバンド分解手段2と、符号量制御装置3と、エントロピ符号生成手段4とを備え、画像符号化規格JPEG2000に代表されるウェーブレット変換を用いたサブバンド符号化により、入力画像を、与えられた符号量上限値内で圧縮符号化し、エントロピ符号に変換するものである。

30

【0061】

サブバンド分解手段2は、入力画像を、ウェーブレット変換を用いた2次元サブバンド分解によって分解するものである。図11(a)に示すように、このサブバンド分解手段2は、入力画像に1次元のウェーブレット変換を水平及び垂直に施すものであり、1回の分解によって入力画像をL1L1、L1H1、H1L1、H1H1に分解する。この分解をさらにL1L1に対して行うことで、L1L1は、L2L2、L2H2、H2L2、H2H2に分解される。このように、サブバンド分解手段2は、入力画像の低周波領域を繰り返して分解することで、入力画像を階層的なサブバンドに分解するものである。

40

【0062】

符号量制御装置3は、サブバンド分解手段2で分解されたサブバンドを特定の大きさのブロック(コードブロック)に分割し、そのブロック毎に量子化のレベル(量子化レベル)を変えて量子化を行いデータの切り捨てを行う際に、符号化歪を最小にする各ブロック毎の量子化レベルを選択するものである。ここでは、量子化レベルとして、図12で説明したビットプレーン符号化で使用するビットプレーンの位置を示すものとする。この符号量制御装置3の構成については後記する。

【0063】

エントロピ符号生成手段4は、符号量制御装置3で各ブロックの量子化レベルによって削

50

減されたデータを、適応的算術符号等のエントロピ符号化を行うものである。このエントロピ符号生成手段 4 によって、符号化データであるエントロピ符号が生成される。

【0064】

(符号量制御装置の構成)

次に、本発明における符号量制御装置の構成について説明する。

図 1 に示すように、符号量制御装置 3 は、コードブロック分割手段 10 と、歪・符号量テーブル生成手段 11 と、符号量基準値リスト生成手段 12 と、作業テーブル生成手段 13 と、逆探索手段 14 と、ビット切り捨て手段 15 と、を備える構成とした。

【0065】

コードブロック分割手段 10 は、サブバンド分解手段 2 によってサブバンドに分解された入力画像を特定の大きさのブロック (コードブロック) に分割するものである (図 11 (b) 参照)。ここでは、コードブロック分割手段 10 は、各サブバンドの画素を P ビットで表現し、さらに各サブバンドを空間的に垂直 m 画素、水平 n 画素の N 個のコードブロック B (j) (j = 1, ..., N) に分割するものとする。これにより、各コードブロックは m × n ビットの P 個のビットプレーンとして生成される。このコードブロックは、歪・符号量テーブル生成手段 11 と、ビット切り捨て手段 15 とに出力される。

【0066】

歪・符号量テーブル生成手段 11 は、コードブロック分割手段 10 で分割されたコードブロックの量子化レベルを変えてエントロピ符号化を行ったときの符号量と、その符号化によって廃棄される歪量とを、各コードブロックと量子化レベルを対応付けた歪・符号量テーブルを生成するものである。なお、このエントロピ符号化は、エントロピ符号生成手段 4 と同一の符号化である。ここで、量子化レベルは、図 12 で説明したビットプレーン符号化におけるビットプレーンを廃棄する基準となる位置 (TP: Truncation Point) を用いることとする。

【0067】

この歪・符号量テーブル生成手段 11 では、歪・符号量テーブルとして、コードブロック B (j) の TP を t (t = 0, ..., P - 1) とし、t より下位のビットプレーンを廃棄してコードブロックを符号化したときの符号量を (1) 式で定義して、符号量テーブル (R (t, j)) とし、(1) 式において廃棄された歪量を、歪量テーブル (D (t, j)) として生成する。すなわち、図 2 に示すように、歪量テーブル (図 2 (a) 参照) 及び符号量テーブル (図 2 (b) 参照) は、P 個のビットプレーンの中で TP の位置を指し示すインデックス t と、コードブロック B (j) を指し示すインデックス j とを引数とする 2 次元の配列となる。この歪量テーブル及び符号量テーブルが請求項に記載の歪・符号量テーブル保持手段に該当し、メモリ等の記憶手段に保持される。

【0068】

符号量基準値リスト生成手段 12 は、与えられた符号量上限値を最大値とする数列を符号量基準値のリスト (符号量基準値リスト K set) として生成するものである。この符号量基準値リスト生成手段 12 は、与えられた符号量上限値を K としたとき、K を最大値とする S (K) 通りの符号量基準値リスト K set を、図 3 に示すように k (1), ..., k (S (K)) の 1 次元配列データとして生成する。例えば、HDTV (High Definition Television) の入力画像を符号化する場合は、符号量基準値リスト K set の数列の個数を 100 ~ 200 個程度とする。この符号量基準値リスト K set が請求項に記載の符号量基準値リスト保持手段に該当し、メモリ等の記憶手段に保持される。

【0069】

作業テーブル生成手段 13 は、歪・符号量テーブル生成手段 11 で生成された歪量テーブル D、符号量テーブル R 及び符号量基準値リスト生成手段 12 で生成された符号量基準値リスト K set を参照して、符号量基準値 k (i) (i = 0, ..., S (K)) 毎に、j 個のコードブロック B (1), ..., B (j) の歪総量が最小となる量子化レベルのリストを設定したときの歪総量を歪総量テーブル F (i, j) とし、そのときの量子化レベル (T

10

20

30

40

50

P)を指し示すインデックスをインデックステーブル $I d x(i, j)$ として生成するものである。この歪総量テーブルF及びインデックステーブル $I d x$ の生成手順については後記する。この歪総量テーブルF及びインデックステーブル $I d x$ が請求項に記載の作業テーブル保持手段に該当し、メモリ等の記憶手段に保持される。

【0070】

逆探索手段14は、歪・符号量テーブル生成手段11で生成された符号量テーブルR、符号量基準値リスト生成手段12で生成された符号量基準値リスト $K s e t$ 及び作業テーブル生成手段13で生成されたインデックステーブル $I d x$ を参照して、N個のコードブロック $B(j)$ ($j = 1, \dots, N$) 毎のTPの位置 $(t(1), \dots, t(N))$ をTPリストTとして決定するものである。

10

【0071】

この逆探索手段14は、作業テーブル生成手段13で生成されたインデックステーブル $I d x$ ですでに求められている $I d x(S(K), N)$ の値、すなわち、コードブロック数をN個としたときの、コードブロック $B(N)$ のTPの位置 $(t(N))$ を初期値として、インデックステーブル $I d x$ から順次 $t(N-1), \dots, t(1)$ を逆探索して求めていくものである。この逆探索手順については後記する。

【0072】

ビット切り捨て手段15は、コードブロック分割手段10で分割されたコードブロック $B(j)$ ($j = 1, \dots, N$) から、逆探索手段14で求められたTPリスト $T(t(1), \dots, t(N))$ に基づいて、各コードブロック $B(j)$ からTPリストTのTPの位置より下位のビットプレーンを廃棄し、残ったビットプレーンをエントロピ符号生成手段4へ出力するものである。

20

これにより、エントロピ符号生成手段4で符号化されたエントロピ符号は、入力画像を、与えられた符号量上限値内で歪量を最小にして圧縮符号化された符号化データとなる。

【0073】

以上、一実施形態に基づいて、符号量制御装置3の構成について説明したが、符号量制御装置3は、コンピュータにおいて各手段を各機能プログラムとして実現することも可能であり、各機能プログラムを結合して符号量制御プログラムとして動作させることも可能である。

【0074】

(符号量制御装置の動作)

30

次に、符号量制御装置3の動作について説明する。

【動作概念】

まず、符号量制御装置3で行う符号量制御動作の概念について説明する。この符号量制御装置3は、N個のコードブロック $B(j)$ ($j = 1, \dots, N$) で前記した(問題A)の解を求めるにあたり、(問題A)を図4に示すようにN通りの部分的な対象に分解し、その部分的な対象における問題を順次解くことにより、最終的に(問題A)の解を得るものである。

【0075】

図4に示したN通りの対象(1~N)について、符号量基準値リスト生成手段12で生成する符号量基準値 $k(i)$ ($i = 1, \dots, S(K)$) を符号量上限値とする、対象 j ($j = 1, \dots, N$) の符号量の割り当てを最適化する問題を、 $A(i, j)$ で表し、この $S(K) \times N$ 通りの問題を以下の(サブ問題A)として定式化する。

40

【0076】

(サブ問題A)

j 個のコードブロック $B(1), B(2), \dots, B(j)$ について、目標とする符号量上限値を $k(i)$ とするとき、 $R(T(j)) = k(i)$ なる条件の下で、 $D(T(j))$ を最小化するTP(Truncation Point)のリスト(TPリスト) $T(j)$ を決定する(R及びDは、(4)式及び(5)式参照)。

ただし、 $T(j) = ((t(1), \dots, t(j)))$ で、 j 個のコードブロック $B(1)$

50

、 $B(2), \dots, B(j)$ にそれぞれ割り当てられる TP のリスト $T(j)$ とする。

(サブ問題 A 終)

【0077】

このように定式化された $S(K) \times N$ 個のサブ問題の中で (サブ問題 A) $A(S(K), N)$ が本来求めたい (問題 A) に該当することになる。

この符号量制御装置 3 では、最も小規模な (サブ問題 A) $A(1, 1)$ から、 i 及び j を順次拡大しながら、すでに解かれた小規模の (サブ問題 A) の解を参照することで、逐次的に問題を解決していき、最終的に求めたい (サブ問題 A) $A(S(K), N)$ 、すなわち (問題 A) を解くものである。

【0078】

次に、図 5 を参照 (適宜図 1 参照) して、(サブ問題 A) を順次解いていく符号量制御装置 3 の動作について説明する。図 5 は、符号量制御装置 3 の概略動作を示すフローチャートである。なお、コードブロック分割手段 10 及びビット切り捨て手段 15 の動作については、従来と同様の動作であるのでここでは説明を省略する。

【0079】

[ステップ S1: 歪・符号量テーブル生成]

まず、符号量制御装置 3 は、ブロック分割手段 10 で $m \times n$ ビットの P 個のビットプレーンとして分割された各コードブロック $B(j)$ ($j = 1, \dots, N$) を、歪・符号量テーブル生成手段 11 によって、量子化レベル (TP) の値 ($t = 0, \dots, P-1$) 毎にエントロピ符号化を行った符号量を符号量テーブル $R(t, j)$ として生成し、その符号化によ

って廃棄される歪量を歪量テーブル $D(t, j)$ として生成する。

【0080】

[ステップ S2: 符号量基準値リスト生成]

また、符号量制御装置 3 は、符号量基準値リスト生成手段 12 によって、与えられた符号量上限値を K としたとき、 K を最大値とする $S(K)$ 通りの符号量基準値のリスト (符号量基準値リスト $Kset$) を、 $Kset = (k(1), \dots, k(S(K))) = K$ として生成する。

【0081】

なお、この符号量基準値リスト $Kset$ は、 $k(i) < k(i+1)$ かつ $k(S(K)) = K$ であればよい。例えば、正数 ε を $K / (S(K) - 1)$ とし、(7) 式に示すように $k(i)$ を直線的に増加する数列とする。

【0082】

$$k(i) = (i - 1) * \varepsilon \quad \dots (7)$$

【0083】

また、符号量基準値リスト $Kset$ は、ある符号量基準値付近を細かく分割する非線型に増加する数列としてもよい。

【0084】

[ステップ S3: 作業テーブル初期値設定]

そして、符号量制御装置 3 は、作業テーブル生成手段 13 によって、(サブ問題 A) を順次、帰納的に解くための作業領域となる $S(K)$ 行 N 列のインデックステーブル Idx 及び歪総量テーブル F に初期値を設定する。

【0085】

ここで、 $Idx(i, j)$ 及び $F(i, j)$ ($i = 1, \dots, S(K)$, $j = 1, \dots, N$) を、インデックステーブル Idx 及び歪総量テーブル F の i 行 j 列の要素とすると、 $F(i, j)$ は、符号量基準値 $k(i)$ で j 個のコードブロック $B(1), \dots, B(j)$ について最適な TP のリスト $T(j)$ を設定したとき、すなわち、(サブ問題 A) $A(i, j)$ を解いたときの歪量総和 $D(T(j))$ を格納する場所となる。また、 $Idx(i, j)$ は、(サブ問題 A) $A(i, j)$ を解いたときのコードブロック $B(j)$ の TP を示す $t(j)$ の値を格納する場所となる。

【0086】

10

20

30

40

50

このステップS3では、コードブロックB(1)に着目して、最も規模の小さい(サブ問題A)A(i, 1) (i = 1, ..., S(K))を解き、インデックステーブルIdx及び歪総量テーブルFの第1列の要素を決定する。この(サブ問題A)A(i, 1) (i = 1, ..., S(K))は、ある符号量基準値k(i)における1個のブロックB(1)に符号量を割り当てる問題であって、求めるべきTPのリストであるT(1)には1個の要素t(1)しか存在しない。従って、(サブ問題A)A(i, 1)の解、すなわち、符号量基準値k(i)の符号量制約の下でコードブロックB(1)に対する最適なTPであるt(1)は、符号量総和R(T) = k(i)、すなわち、R(t(1), 1) = k(i)によって一意に決定されることになる。この場合、必ずしもR(t(1), 1) = k(i)が正確に成立するt(1)が存在するとは限らないため、ここでは、k(i)に「最も近い」符号量R(t, 1)となるt、すなわちt(1)を決定し、このt(1)の値をインデックステーブルIdx(i, 1)に格納する。

10

【0087】

そして、歪総量テーブルFのF(i, 1)には、(サブ問題A)A(i, 1)での歪量、すなわち、インデックステーブルIdxのIdx(i, 1)におけるコードブロックB(1)に発生する符号化歪量を、歪量テーブルDから歪量D(Idx(i, 1), 1)の値を読み込んで格納する。

【0088】

ここで「最も近い」について定義しておく。例えば符号量をrとし、符号量基準値k(i) (i = 1, ..., S(K))としたとき、近さの尺度を|k(i) - r| (k(i)とrとの差分の絶対値)とすると、k(i)は(7)式に示したようにiに関して単調増加(k(i-1) < k(i))するものであり、k(i-1) < r < k(i)であって、rがk(i-1)とk(i)とのちょうど中間に位置する場合、すなわちr - k(i-1) = k(i) - rであるとすると、rに「最も近い」値はk(i-1)とk(i)との2個になり、ある符号量rに「最も近い」符号量基準値が特定できないことになる。

20

【0089】

そこで、ここではk(i) ≤ rで、かつ、i' ≠ iである全てのi'について|r - k(i)| ≤ |r - k(i')|とすることで、ある符号量rに「最も近い」符号量基準値k(i)のiを特定するものとする。なお、このiを特定する方法はこれ以外にも種々あり、例えば、k(i) ≥ rで、かつ、i' ≠ iである全てのi'について|r - k(i)| ≤ |r - k(i')|とすることも可能である。

30

【0090】

以上の作業テーブル初期値設定(ステップS3)の動作を図6のフローチャートに示す。図6に示すように、作業テーブル初期値設定(ステップS3)は、まず変数iを1に初期化する(ステップS30)。そして、符号量基準値k(i)に「最も近い」符号量R(t, 1)となるコードブロックB(1)に対するTPの値をtとする(ステップS31)。なお、このステップS31において、argument {条件|パラメータリスト}は「パラメータリスト」から「条件」を満たすパラメータを取り出す操作を表すものとする。また、「条件」における「≡」記号は「最も近い」を表す記号とする。

【0091】

40

そして、このステップS31で取得したTPの値tを、インデックステーブルIdx(i, 1)に格納し(ステップS32)、歪量D(t, 1)すなわちD(Idx(i, 1), 1)を読み込んで、歪総量テーブルFのF(i, 1)に格納する(ステップS33)。そして、変数iが符号量基準値の数(S(K))かどうかを判定し(ステップS34)、等しい場合(Yes)は処理を終了し、異なる場合(No)は変数iの値を1加算して(ステップS35)、ステップS31へ戻って処理を続ける。

【0092】

以上の動作によって、符号量基準値k(i) (i = 1, ..., S(K))において、インデックステーブルIdxのIdx(i, 1)には、コードブロックB(1)に対して、k(i)に「最も近い」符号量を与えるTPの値が格納される。また、歪総量テーブルFのF

50

($i, 1$)には、コードブロックB(1)に対して、 $k(i)$ に「最も近い」符号量を与えるTPにおけるコードブロックB(1)の歪量が格納される。

これにより、インデックステーブルIdx及び歪総量テーブルFにおける第1列目のS(K)個の値が埋められ、(サブ問題A)A($i, 1$)($i = 1, \dots, S(K)$)が解かれたことになる。

【0093】

〔ステップS4：作業テーブル設定〕

次に符号量制御装置3は、作業テーブル生成手段13によって、ステップS3で解かれた(サブ問題A)A($i, 1$)の結果を初期値として、コードブロックB(i), ..., B($j-1$)に関する(サブ問題A)A($i, j-1$)を参照して、(サブ問題A)A(i, j)を解いていく。すなわち、インデックステーブルIdx及び歪総量テーブルFですでに確定している第($j-1$)列のS(K)個の値を参照して、第j列の値を帰納的に決定していく。

【0094】

ここで作業テーブル設定(ステップS4)の動作を図7のフローチャートを参照して説明する。

図7に示すように、作業テーブル設定(ステップS4)は、まず、コードブロックB(j)のインデックスである変数jを2に初期化し(ステップS40)、符号量基準値 $k(i)$ のインデックスである変数iを1に初期化する(ステップS41)。ここで変数jを2に初期化するのは、作業テーブル初期値設定(ステップS3)において、コードブロックB(1)におけるインデックステーブルIdxは、すでに設定されているためである。

【0095】

そして、(サブ問題A)A(i, j)において、TPのリストであるT(j)の最後の要素 $t(j)$ を t としたとき(コードブロックB(j)のTPを t としたとき)のコードブロックB(1), ..., B(j)の歪総量を算出する(ステップS42)。

ここでは、まず、符号量 r から符号量基準値リストKsetの要素のインデックスへの写像 ψ を(8)式で定義しておく。

【0096】

$$\psi(r) = \text{argument} \{ r \approx k(i) \mid i \in \{1, \dots, S(K)\} \} \quad \dots (8)$$

【0097】

この(8)式において、 $\text{argument} \{ \text{条件} \mid \text{パラメータリスト} \}$ は「パラメータリスト」から「条件」を満たすパラメータを取り出す操作を表すものとし、「条件」における「 \approx 」記号は「最も近い」を表す記号とする。

そして、このステップS42のコードブロックB(1), ..., B(j)の歪総量 $f(i, j; t)$ の算出には、(9)式を用いる。

【0098】

$$f(i, j; t) = D(t, j) + F(\psi(k(i) - R(t, j)), j-1) \quad \dots (9)$$

【0099】

この(9)式において、($k(i) - R(t, j)$)は、(サブ問題A)A(i, j)に割り当てられた符号量基準値 $k(i)$ から、コードブロックB(j)の符号量 $R(t, j)$ を引いた量で、ここでは残余符号量と呼ぶこととする。また、 $\psi(k(i) - R(t, j))$ は、(8)式に示したように、残余符号量($k(i) - R(t, j)$)に「最も近い」符号量基準値リストKsetの要素のインデックスを表している。従って、歪総量テーブルFの $F(\psi(k(i) - R(t, j)), j-1)$ は、(サブ問題A)A($\psi(k(i) - R(t, j)), j-1$)を解いたときの歪総量となる。

【0100】

10

20

30

40

50

そして、(9)式は、この(サブ問題A) $A(\psi(k(i) - R(t, j), j - 1))$ を解いたときの歪総量に、コードブロック $B(j)$ の歪量 $D(t, j)$ を加算したもので、(サブ問題A) $A(i, j)$ において、TPのリストである $T(j)$ の最後の要素 $t(j)$ を t としたときのコードブロック $B(1), \dots, B(j)$ の歪総量となる。このように、(9)式で算出した歪総量が最小となるTPの値 t を(10)式を用いて取得する(ステップS43)。

【0101】

$t \leftarrow \text{argument} \{ \text{minimize } f(i, j; t) \mid t \in \{0, \dots, P-1\} \} \quad \dots (10)$

10

【0102】

この(10)式は、歪総量 $f(i, j; t)$ がパラメータ t によって変化するとき最小値(minimize)となる t の値を取り出すことを表している。ここで、歪総量 $f(i, j; t)$ の要素は最大 $P(0, \dots, P-1)$ 個あり、その中の最小値が、(サブ問題A) $A(i, j)$ を解いたときの歪総量である。従って、その歪総量が最小値となるときのTPの値 t を取得すれば、この t の値が、(サブ問題A) $A(i, j)$ の解であるTPのリスト $T(j)$ における最後の要素 $t(j)$ となる。

【0103】

そして、ステップS43で取得したTPの値 t を、インデックステーブル $Idx(i, j)$ に格納し(ステップS44)、そのときの t の値における歪総量 $f(i, j; t)$ を、歪総量テーブル F の $F(i, j)$ に格納する(ステップS45)。そして、変数 i が符号量基準値の数 $S(K)$ かどうかを判定し(ステップS46)、異なる場合(No)は変数 i の値を1加算して(ステップS47)、ステップS42へ戻って処理を続ける。

20

【0104】

そして、変数 i が符号量基準値の数 $S(K)$ と等しい場合(Yes)は、変数 j がコードブロックの数 N かどうかを判定し(ステップS48)、等しい場合(Yes)は処理を終了し、異なる場合(No)は変数 j の値を1加算して(ステップS49)、ステップS41へ戻って処理を続ける。

【0105】

以上の動作によって、インデックステーブル Idx 及び歪総量テーブル F における第2列目以降のコードブロック $B(j)$ ($j = 2, \dots, N$) の値が $S(K)$ 個ずつ順に埋められ、(サブ問題A) $A(i, j)$ を解いたときの、インデックステーブル Idx の値である $Idx(i, j)$ と、歪総量テーブル F の値である $F(i, j)$ が決定される。このインデックステーブル Idx 及び歪総量テーブル F の要素が逐次的に決定される順序を図9に示す。図9(a)には、インデックステーブル Idx の決定順序を、図9(b)には、歪総量テーブル F の決定順序を模式的に示している。インデックステーブル Idx 及び歪総量テーブル F 共に、第1列目(コードブロック $B(1)$)の符号量基準値 $k(i)$ のインデックス $i = 1$ から $i = S(K)$ までが初期値として設定された後に、第2列目のコードブロック $B(2)$ から順番にコードブロック $B(N)$ までの値が決定される。

30

【0106】

このように、帰納的に(サブ問題A)を解いていくことで、(サブ問題A) $A(S(K), N)$ における歪総量テーブル F の値である $F(S(K), N)$ と、インデックステーブル Idx の値である $Idx(S(K), N)$ とを求めることができる。

【0107】

[ステップS5：逆探索]

次に符号量制御装置3は、作業テーブル生成手段13によって、ステップS4で設定されたインデックステーブル Idx の $S(K)$ 行 N 列のテーブルである $Idx(i, j)$ ($i = 1, \dots, S(K), j = 1, \dots, N$) を用いて、 $Idx(S(K), N)$ 、すなわち、コードブロックの数が N 個の場合のコードブロック $B(N)$ におけるTPの値 $t(N)$ を初期値として、逆探索により $t(N-1), \dots, t(1)$ を順次決定する。

40

50

【0108】

ここで、逆探索（ステップS5）の動作を図8のフローチャートを参照して説明する。図8に示すように、逆探索（ステップS5）は、まず、符号量基準値 $k(i)$ のインデックスである変数 i を符号量基準値の数 $(S(K))$ に初期化し（ステップS50）、コードブロック $B(j)$ のインデックスである変数 j をコードブロックの数 (N) に初期化する（ステップS51）。

【0109】

そして、コードブロック $B(j)$ に割り当てるTPの値である $Idx(i, j)$ をインデックステーブル Idx から取得し、 $t(j)$ に設定する（ステップS52）。このステップS52は、ある $Idx(i, j)$ が定まっているとすると、符号量基準値 $k(i)$ をコードブロック $B(1), \dots, B(j)$ に割り当てる（サブ問題A） $A(i, j)$ はすでに解かれており、 $Idx(i, j)$ は、最後のコードブロック $B(j)$ に割り当てるTPの値である $t(j)$ となる。

【0110】

この結果、符号量基準値 $k(i)$ から、コードブロック $B(j)$ に割り当てる符号量 $R(t, j)$ を引いた量、すなわち、残余符号量 $k(i) - R(t(j), j)$ がコードブロック $B(1), \dots, B(j-1)$ に割り当てるべき符号量となる。そこで、このコードブロック $B(j)$ の残余符号量 $k(i) - R(t(j), j)$ に「最も近い」符号量基準値のインデックスを $\psi(k(i) - R(t(j), j))$ で算出し、新たに変数 i に代入する（ステップS53）。

【0111】

そして、コードブロック $B(j)$ のインデックスである変数 j から1減算し（ステップS54）、変数 j が0になったかどうかを判定する（ステップS55）。ここで変数 j が0になった場合（Yes）は、処理を終了し、変数 j が0でない場合（No）は、ステップS52へ戻って処理を続ける。

【0112】

以上の動作を図10を参照してさらに説明する。図10は、逆探索手段14における逆探索動作を模式的に表した図である。インデックステーブル Idx の値である $Idx(S(K), N)$ 、すなわちコードブロック $B(N)$ のTPの値 $(t(N))$ は、すでにステップS4によって求められている（A1に書き込まれている）。次に、符号量上限値 K から、 $t(N)$ でコードブロック $B(N)$ を符号化したときの符号量 $(R(t(N), N))$ を引いた残余符号量 $(K - R(t(N), N))$ に「最も近い」符号量基準値リスト $Kset$ の要素のインデックスを i とする。すなわち、 $K - R(t(N), N) \approx k(i)$ となるインデックス i を求める。そして、インデックステーブル Idx で、このインデックス i に該当する（A2に書き込まれている）コードブロック $B(N-1)$ の値 $(Idx(i, N-1))$ をコードブロック $B(N-1)$ のTPの値 $(t(N-1))$ とする。

【0113】

さらに、符号量基準値 $k(i)$ から、 $t(N-1)$ でコードブロック $B(N-1)$ を符号化したときの符号量 $(R(t(N-1), N-1))$ を引いた残余符号量 $(K - R(t(N-1), N-1))$ に「最も近い」符号量基準値リスト $Kset$ の要素のインデックスを i' とする。すなわち、 $K - R(t(N-1), N-1) \approx k(i')$ となるインデックス i' を求める。そして、インデックステーブル Idx で、このインデックス i' に該当する（A3に書き込まれている）コードブロック $B(N-2)$ の値 $(Idx(i', N-2))$ をコードブロック $B(N-2)$ のTPの値 $(t(N-2))$ とする。このように、順次TPの値を $t(1)$ まで逆探索する。

【0114】

以上の動作によって、 N 個のコードブロック $B(j)$ のTPのリスト $T = (t(1), \dots, t(N))$ は、すでに求まっているインデックステーブル Idx を参照（逆探索）することで、 $t(N), \dots, t(1)$ の順番に決定される。

以上のステップS1～S5の動作により、与えられた符号量上限値 K の下で歪量総和 D （

10

20

30

40

50

T) ((5) 式参照) を最小化するコードブロック B (j) (j = 1, ..., N) についての TP (Truncation Point) のリスト (TP リスト) T を逐次的に決定することができる。

【0115】

このステップ S1 ~ S5 の動作で明らかなように、本手法によれば (問題 A) は、S (K) * N * P のオーダの演算で解くことが可能である。例えば、符号量基準値の数を 100 (S (K) = 100)、コードブロックの数を 64 (N = 64)、各コードブロックのビットプレーンの数を 16 (P = 16) とすると、51200 回の演算ステップで動作させることが可能で、例えば、1 ステップあたり 100 ナノ秒 (ns) の演算時間としても、1/30 秒のフレームレートで本手順を動作させることができる。これにより、ソフトウェアによる符号量制御を実現することができる。

10

【0116】

なお、本手法において、量子化レベルをビットプレーンの位置を示す TP (Truncation Point) として説明したが、この量子化レベルは、サブバンド符号化で用いられるビットプレーン符号化に限定されるものではなく、ブロック単位で符号量と、歪量とをパラメータ制御するような符号化方式においても、そのパラメータを量子化レベルとみなすことで実現することが可能である。例えば、DCT (Discrete Cosine Transform: 離散コサイン変換) ベースのブロック符号化においては、ブロック単位に量子化レベルを決定すればよい。

【0117】

20

また、JPEG2000 では、1 つのビットプレーンが 3 種類のパスに分類されてエントロピ符号化されるが、本手法においては、パス毎に別のビットプレーンとみなすことで前記した動作と同様に扱うことが可能である。

【0118】

【発明の効果】

以上説明したとおり、本発明に係る符号量制御方法、符号量制御装置及び符号量制御プログラムでは、以下に示す優れた効果を奏する。

【0119】

請求項 1、請求項 2、請求項 4 又は請求項 5 に記載の発明によれば、情報量を廃棄 (量子化) することで、所定の符号量上限値の範囲内で圧縮符号化を行う画像符号化において、歪量を最小化する符号をテーブル検索及び四則演算による短いステップの演算で決定することができる。

30

これによって、汎用の CPU であっても、動画像を実時間で高画質に圧縮符号化を行うことが可能になり、低コストで符号化装置を実現することができる。

【0120】

請求項 3 に記載の発明によれば、量子化を行うレベル (量子化レベル) をビットプレーン符号化におけるビットプレーンを廃棄する基準位置とすることで、例えば、ビットプレーン符号化によって量子化を行う JPEG2000 等で、画像をブロックに分割し、そのブロックのビットプレーンを廃棄する位置を各ブロック毎に制御することが可能になる。

【図面の簡単な説明】

40

【図 1】本発明の実施の形態に係る符号量制御装置の全体構成を示すブロック図である。

【図 2】本発明の実施の形態に係る歪量テーブル及び符号量テーブルの構造を説明するための説明図である。

【図 3】本発明の実施の形態に係る符号量基準値リストの構造を説明するための説明図である。

【図 4】本発明の実施の形態に係る (問題 A) の解を求めるためにコードブロックを複数の対象に分解する作業を説明するための説明図である。

【図 5】本発明の実施の形態に係る符号量制御動作の概略を示すフローチャートである。

【図 6】本発明の実施の形態に係る作業テーブル初期設定の動作を示すフローチャートである。

50

【図 7】本発明の実施の形態に係る作業テーブル設定の動作を示すフローチャートである。

【図 8】本発明の実施の形態に係る逆探索の動作を示すフローチャートである。

【図 9】本発明の実施の形態に係るインデックステーブル及び歪総量テーブルを設定する順序を視覚的に表した模式図である。

【図 10】本発明の実施の形態に係る逆探索動作を模式的に表した模式図である。

【図 11】サブバンド符号化におけるサブバンド分解とコードブロック分割を説明するための説明図である。

【図 12】ビットプレーン符号化の動作の概要を説明するための説明図である。

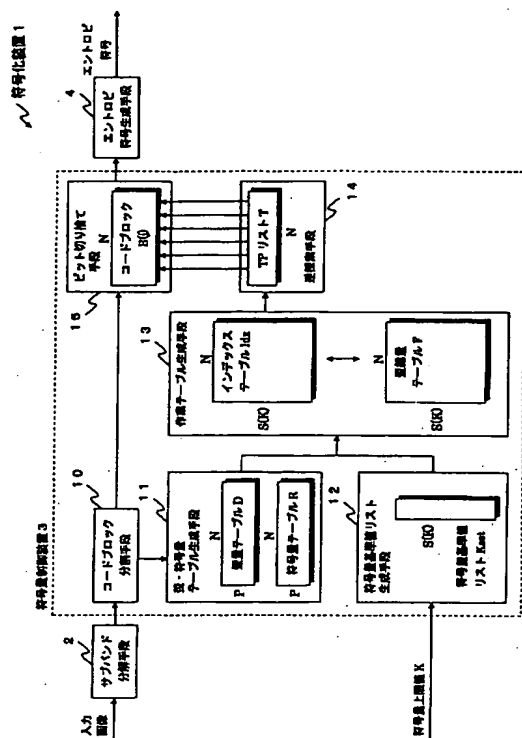
【図 13】ビットプレーン符号化における Truncation point と符号量及び歪量との関係を説明するための説明図である。 10

【図 14】ビットプレーンの位置（大きさ）と、歪量及び符号量の関係を説明するための説明図である。

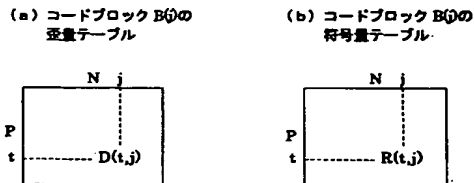
【符号の説明】

- 1 ……符号化装置
- 2 ……サブバンド分解手段
- 3 ……符号量制御装置
- 4 ……エントロピ符号生成手段
- 10 ……コードブロック分割手段
- 11 ……歪・符号量テーブル生成手段
- 12 ……符号量基準値リスト生成手段
- 13 ……作業テーブル生成手段
- 14 ……逆探索手段
- 15 ……ビット切り捨て手段
- D ……歪量テーブル（歪・符号量テーブル保持手段）
- R ……符号量テーブル（歪・符号量テーブル保持手段）
- K s e t ……符号量基準値リスト（符号量基準値リスト保持手段）
- I d x ……インデックステーブル（作業テーブル保持手段）
- F ……歪総量テーブル（作業テーブル保持手段）

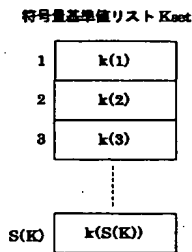
【 図 1 】



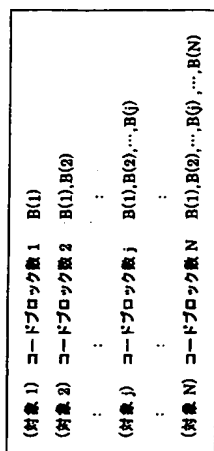
【圖 2】



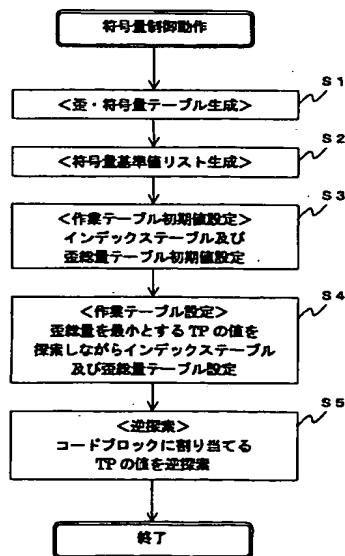
【図 3】



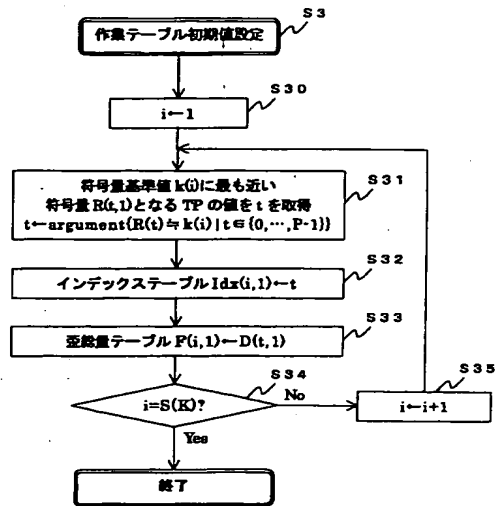
【図 4】



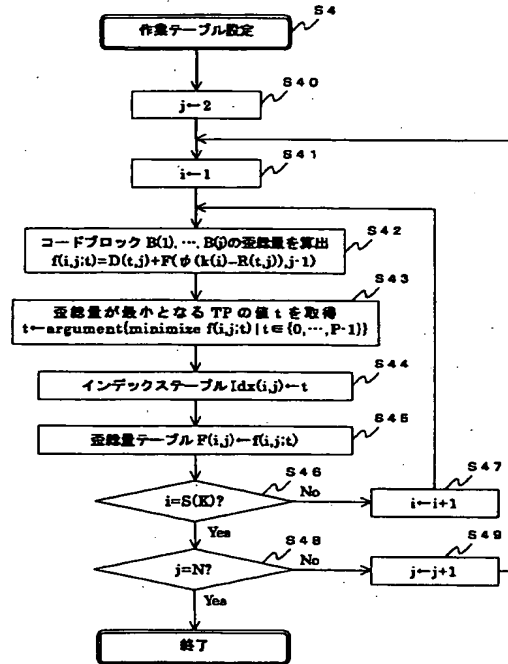
【 図 5 】



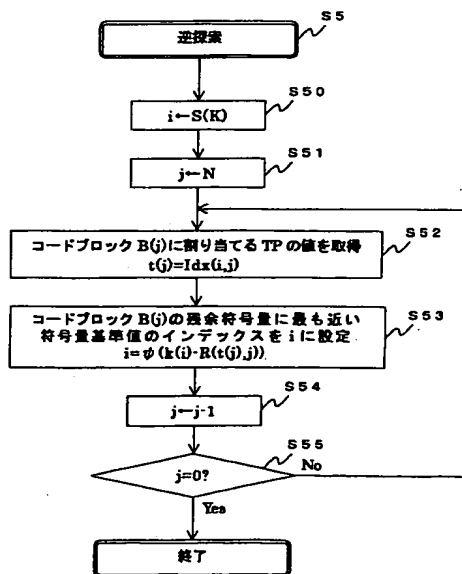
【図 6】



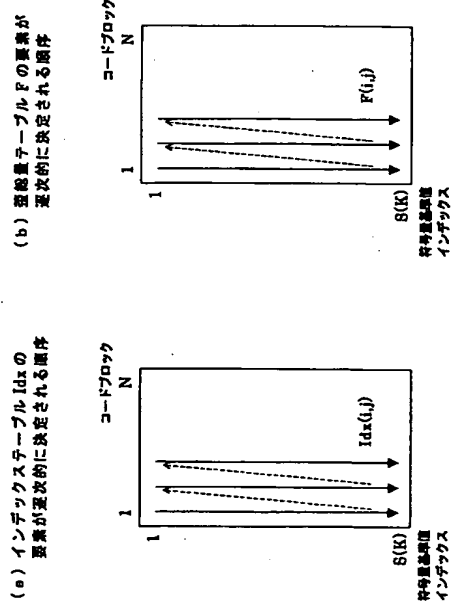
【図 7】



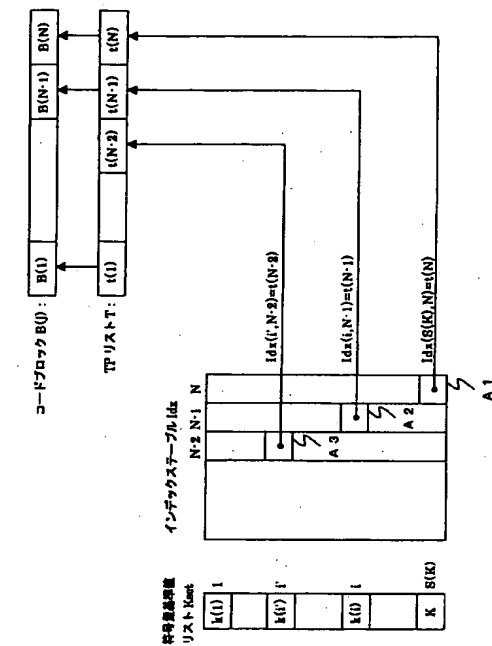
【図 8】



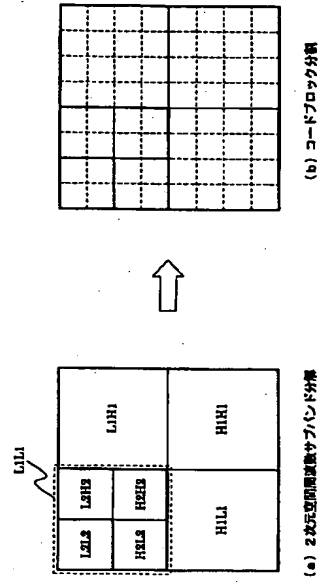
【図 9】



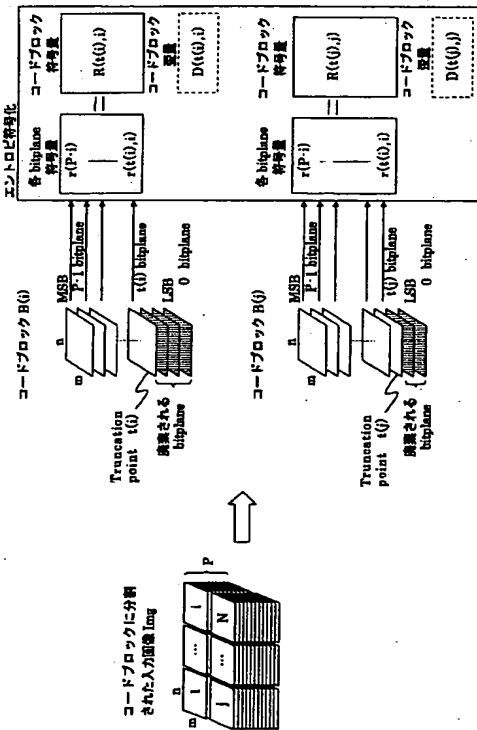
【図 10】



【図 11】



【図 12】



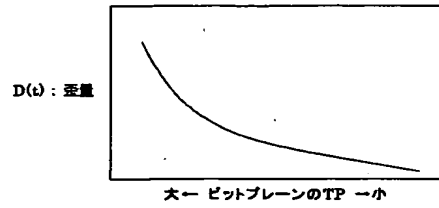
【図 13】

Truncation point (TP) と、符号量及び量との関係

ビットプレーンの TP の位置 t	R(t) : 符号量	D(t) : 量
0	$r(0)+r(1)+r(2)+\dots+r(16)$	0
1	$r(1)+r(2)+\dots+r(16)$	$d(0)$
2	$r(2)+\dots+r(16)$	$d(0)+d(1)$
:	:	:
t	$r(t)+\dots+r(16)$	$d(0)+d(1)+\dots+d(t-1)$
:	:	:
14	$r(14)+r(15)$	$d(0)+d(1)+\dots+d(13)$
15 (最後の符号プレーン)	0	$d(0)+d(1)+\dots+d(13)+d(14)$

【図 14】

(a) ビットプレーンの TP と歪量との関係



(b) ビットプレーンの TP と符号量との関係

